

AD-A205 174

HPC E&L COPY

ARO 24923.3-MA

(2)

# UNIVERSITY OF COLORADO

A New Modified Cholesky Factorization

Robert B. Schnabel and Elizabeth Eskow

December 1988

DEPARTMENT OF COMPUTER SCIENCE  
CAMPUS BOX 430  
UNIVERSITY OF COLORADO, BOULDER  
BOULDER, COLORADO 80309-0430

Technical Report

DTIC  
SELECTED  
FEB 17 1989  
S D  
C P

DISTRIBUTION STATEMENT A  
Approved for public release  
Distribution Unlimited

89 2 16 022

(2)

## A New Modified Cholesky Factorization

Robert B. Schnabel and Elizabeth Eskow

December 1988

Department of Computer Science  
Campus Box 430  
University of Colorado,  
Boulder, Colorado, 80309 USA

DTIC  
ELECTED  
FEB 17 1989  
S D  
D<sup>as</sup>

This research was supported by ARO grant DAAL-03-88-0086, NSF grant CCR-8702403, and NSF cooperative agreement DCR-8420944.

DISTRIBUTION STATEMENT A  
Approved for public release  
Distribution Unlimited

Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author and do not necessarily reflect the views of the National Science Foundation.

The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

Accession For	
NTIS	CRA&I <input checked="" type="checkbox"/>
DTIC	TAB <input type="checkbox"/>
Unannounced <input type="checkbox"/>	
Justification _____	
By _____	
Distribution / _____	
Availability Codes	
Dist	Avail and / or Special
A-1	

### **Abstract**

The modified Cholesky factorization of Gill and Murray plays an important role in optimization algorithms. Given a symmetric but not necessarily positive definite matrix  $A$ , it computes a Cholesky factorization of  $A+E$ , where  $E=0$  if  $A$  is safely positive definite, and  $E$  is a diagonal matrix chosen to make  $A+E$  positive definite otherwise. The factorization costs only a small multiple of  $n^2$  operations more than the standard Cholesky factorization. We present a new algorithm that has these same properties, but for which the theoretical bound on  $\|E\|$  is substantially smaller. It is based upon two new techniques, the use of Gershgorin bounds in selecting the elements of  $E$ , and a new way of monitoring positive definiteness. In extensive computational tests on indefinite matrices, the new factorization virtually always produces smaller values of  $\|E\|$  than the existing method, without impairing the conditioning of  $A+E$ . In some cases the improvements are substantial. The new factorization may prove useful in optimization algorithms.

## 1. Introduction

The modified Cholesky factorization was introduced by Gill and Murray [1974], and subsequently refined by Gill, Murray, and Wright [1981]. Given a symmetric, not necessarily positive definite matrix  $A \in R^{n \times n}$ , it calculates a Cholesky (i.e.  $LL^T$ , or equivalently  $LDL^T$ ) factorization of  $A+E$ , where  $E$  is 0 if  $A$  is safely positive definite, and  $E$  is a non-negative diagonal matrix for which  $A+E$  is positive definite otherwise. When  $A$  is not positive definite, there is an a priori error bound on how large  $E$  can be as a function of  $A$ ; the practical intent is that  $E$  not be much larger than is necessary to make  $A+E$  positive definite. The factorization uses only about  $n^2/2$  more operations than the normal Cholesky factorization, which costs approximately  $\frac{n^3}{6}$  each multiplication and addition.

The modified Cholesky factorization has become very important in optimization algorithms. Its primary use is in line search methods for unconstrained optimization, where it is used to generate a descent search direction when the Hessian matrix is not positive definite (see e.g. Gill, Murray, and Wright [1981]). It is also used in line search methods for constrained optimization problems (Gill, Murray, and Wright [1981]), and in some trust region methods (Dennis and Schnabel [1983]).

This paper presents a new modified Cholesky factorization algorithm that is intended for the same purposes as the current method. The new algorithm still costs only a small multiple of  $n^2$  operations more than the standard Cholesky factorization. It possesses a much smaller a priori bound on the size of the diagonal matrix  $E$ , and in extensive computational tests,  $\|E\|$  almost never is larger, and in many cases is considerably smaller, than that generated by the algorithm of Gill, Murray, and Wright. In fact, when  $A$  is not positive definite,  $\|E\|$  is usually close enough to the negative of the smallest eigenvalue of  $A$  that the new algorithm may be a useful, inexpensive way to estimate this eigenvalue.

The remainder of this paper is organized as follows. Section 2 contains a brief summary of the motivation and uses for the modified Cholesky factorization in optimization algorithms. Section 3 summarizes the goals of this factorization and the basic challenges that it presents, and section 4 briefly describes the Gill, Murray, and Wright [1981] algorithm.

In Section 5 we present the new algorithm. It contains two main novel features, the use of Gershgorin bounds in determining both the pivot sequence and the elements of  $E$ , and a new two-phase strategy for determining when a matrix is not positive definite and needs to be perturbed. In Section 6 we present the results of an extensive computational comparison of the behavior of the new and old factorizations on indefinite test matrices of dimensions 25 to 75. Section 7 contains some brief conclusions.

Throughout the paper we consider the Cholesky factorization, i.e the factorization into  $LL^T$ , where  $L$  is lower triangular, as opposed to the  $LDL^T$  factorization, where  $L$  is unit lower triangular (ones on the diagonal) and  $D$  is a positive diagonal matrix. The conclusions of the paper are true for either factorization. We use the Cholesky because we believe it makes the exposition simpler. We use the version of the Cholesky factorization that makes a rank one change to the remaining submatrix at each iteration (analogous to Gaussian elimination), rather than the version that delays the changes to any element until it is in the pivot column (analogous to Crout reduction). The use of the first version will be seen in Section 5 to be important to our algorithm.

An important piece of notation, used throughout the paper, is that we use  $A_j$  to denote the principal submatrix that remains to be factored at the start of the  $j^{th}$  iteration of the factorization. Thus,  $A_j$  is an  $n+1-j \times n+1-j$  matrix, consisting of the values that reside in rows and columns  $j$  through  $n$  at the start of the  $j^{th}$  iteration. For consistency,  $A_1$  is the original matrix  $A$ . This notation is expanded in Section 3.

## 2. The Use of the Modified Cholesky Factorization in Optimization Algorithms

The modified Cholesky factorization was introduced by Gill and Murray [1974] in the context of a line search method for solving the unconstrained optimization problem

$$\underset{x \in R^n}{\text{minimize}} \quad f : R^n \rightarrow R .$$

Unconstrained optimization methods generally base each iteration upon the quadratic model of  $f(x)$  around the current iterate  $x_c$

$$m(x_c + d) = f(x_c) + \nabla f(x_c)^T d + \frac{1}{2} d^T H_c d , \quad (2.1)$$

where  $H_c$  is the Hessian matrix  $\nabla^2 f(x_c)$  or a symmetric approximation to it. If  $H_c$  is positive definite, then the step  $d_c = -H_c^{-1} \nabla f(x_c)$  is the minimizer of (2.1) and also a descent direction for  $f(x)$ , so that a satisfactory next iterate  $x_+$  always can be found by choosing  $x_+ = x_c - \lambda_c d_c$  for some  $\lambda_c > 0$ . If  $H_c$  has one or more negative eigenvalues, however, then the model (2.1) is unbounded below, and the direction  $d_c = -H_c^{-1} \nabla f(x_c)$  may or may not be a descent direction for  $f(x)$ . In this case, Gill and Murray [1974] suggested calculating  $d_c = -(H_c + E_c)^{-1} \nabla f(x_c)$  as the search direction, where  $H_c + E_c$  is positive definite, and again choosing  $x_+ = x_c - \lambda_c d_c$  for some  $\lambda_c > 0$  by a line search procedure. By standard convergence results, if  $\|H_c\|$  is uniformly bounded above,  $\|E_c\|$  is bounded above as a function of  $\|H_c\|$ , and the condition number of  $H_c + E_c$  is uniformly bounded above, then the sequence of iterates generated by a standard line search method that uses such search directions will be globally convergent in the sense that the limit of the sequence of gradients converges to zero. If  $E = 0$  when  $H$  is positive definite, then the method will also be quadratically convergent in the neighborhood of a strong local minimizer. (See Dennis and Schnabel [1983] for a summary of these results.)

The algorithm of Gill and Murray [1974] for choosing  $E_c$  satisfies all the aforementioned conditions on  $E_c$ . It also is very efficient in that it calculates either the Cholesky factorization of

$H_c$  if it is positive definite, or the Cholesky factorization of  $H_c + E_c$  otherwise, at barely a higher total cost than a standard Cholesky factorization, without knowing a priori whether  $H_c$  is positive definite or not. For these reasons, it has become a standard technique in line search methods for unconstrained optimization problems. The algorithm was refined somewhat subsequent to its introduction, and a modern version, that has performed very well, is given in Gill, Murray, and Wright [1981].

The modified Cholesky factorization is also used in some line search methods for solving constrained optimization problems. Some algorithms for solving such problems also generate a sequence of unconstrained quadratic models, and if the Hessian of any such model is not positive definite, the same techniques are applicable. For more details, see e.g. Gill, Murray, and Wright [1981].

Some trust region methods for optimization problems also use the modified Cholesky factorization. While we will not elaborate upon these methods here, in some of them it is useful to have an upper bound on the most negative eigenvalue  $\lambda_1$  of  $H_c$ , and the norm of the matrix  $E_c$  from the modified Cholesky factorization serves this purpose. Dennis and Schnabel [1983] discuss trust region methods that incorporate the modified Cholesky factorization, and Shultz, Schnabel, and Byrd [1985] show how to construct efficient and globally convergent trust region methods if a satisfactory upper bound on  $\lambda_1$  is available. The methods described in this paper produce bounds that are satisfactory in this sense.

Our general reasons for pursuing a new modified Cholesky factorization algorithm were given in Section 1 and are elaborated further at the end of Section 4. In addition, from an optimization perspective, the new method may lead to a new efficient and simple implementation of trust region methods. We discuss this possibility briefly in Section 7.

### 3. Goals and Challenges of the Modified Cholesky Factorization

Given a matrix  $A \in R^{n \times n}$  that is symmetric but not necessarily positive definite, the objective of the modified Cholesky factorization is to construct a Cholesky ( $LL^T$ ) factorization of a positive definite matrix  $A+E$ , where  $E$  is a non-negative diagonal matrix. More specifically, the factorization has the following four goals : 1) If  $A$  is safely positive definite,  $E$  should equal 0 ; 2) If  $A$  is indefinite,  $\|E\|$  should not be much greater than  $-\lambda_1(A)$ , where  $\lambda_1(A)$  is the most negative eigenvalue of  $A$  ; 3)  $A+E$  should be a reasonably well conditioned matrix, and 4) the cost of the factorization should only be a small multiple of  $n^2$  operations more than the cost of the normal Cholesky algorithm.

One obvious way to select  $E$  would be to find  $\lambda_1(A)$ , and, if  $\lambda_1(A) < 0$ , let  $E$  equal  $[-\lambda_1(A) + \epsilon] I$ , for some small positive  $\epsilon$ . This would satisfy the first 3 goals, but the expense of finding the eigenvalues of a matrix exceeds the cost requirements specified in our final goal by at least an order of magnitude. Thus the major challenge in developing a modified Cholesky factorization is to satisfy the first 3 goals while not increasing the cost by more than  $O(n^2)$ . Among other things, this implies that a one pass algorithm is essential.

There is a basic tradeoff in deciding upon the size of each of the diagonal elements of the matrix  $E$ . Let the  $n+1-j \times n+1-j$  principal submatrix remaining to be factored at the  $j^{th}$  iteration, consisting of the current elements in rows  $j$  through  $n$  and columns  $i$  through  $n$ , be denoted

$$A_j = \begin{bmatrix} \alpha_j & a_j^T \\ a_j & \hat{A}_j \end{bmatrix},$$

where  $\alpha_j \in R$  is the current  $j^{th}$  diagonal element,  $a_j \in R^{n-j}$  is the current vector of elements in column  $j$  below the diagonal, and  $\hat{A}_j \in R^{(n-j) \times (n-j)}$ . (We will use the convention that the subscripts of the elements in the vector  $a_j$  are  $i = j+1$  through  $n$ , so that  $(a_j)_i = A_{ij}$ ,  $i=j+1, \dots, n$ ). Then at the  $j^{th}$  iteration, the normal Cholesky factorization algorithm computes  $L_{jj} = \sqrt{\alpha_j}$ ,  $L_{ij} =$

$(a_j)_i/L_{jj}$ ,  $i=j+1, \dots, n$ , and (assuming the changes to the remaining elements are not deferred)

$$A_{j+1} = \hat{A}_j - \frac{a_j a_j^T}{\alpha_j}$$

In the modified Cholesky factorization, the computations are instead  $L_{jj} = \sqrt{\alpha_j + \delta_j}$ ,  $L_{ij} = (a_j)_i/L_{jj}$ ,  $i=j+1, \dots, n$ , and

$$A_{j+1} = \hat{A}_j - \frac{a_j a_j^T}{\alpha_j + \delta_j},$$

where  $\delta_j$  is greater than or equal to zero and is the  $j^{th}$  diagonal element of the matrix  $E$ . The tradeoff between making  $\delta_j$  large or small leads to the following dilemma. If  $\alpha_j$  is negative and  $\delta_j$  is chosen so small that  $\alpha_j + \delta_j$  is barely greater than 0, then  $\frac{a_j a_j^T}{\alpha_j + \delta_j}$  will be large, and  $A_{j+1}$  will have large negative eigenvalues, implying that the elements of  $E$  in some remaining iterations will need to be large. On the other hand, if  $\delta_j$  is large, then we have already added a large amount to the diagonal. The challenge lies in adding the appropriate amount to the diagonal of  $A$  at the appropriate time in the algorithm. This requires that the algorithm consider more information than just the value of  $\alpha_j$  in choosing  $\delta_j$ . It will be seen in Sections 4 and 5 that considering the values of  $a_j$  as well as  $\alpha_j$  is sufficient to produce effective modified Cholesky factorization algorithms in both theory and practice.

#### 4. The Modified Cholesky Factorization of Gill, Murray, and Wright

Gill, Murray, and Wright [1981] give a modified Cholesky factorization algorithm that is designed to satisfy the four goals stated at the start of Section 3. Given a symmetric but not necessarily positive definite matrix  $A \in R^{n \times n}$ , it computes an  $LDL^T$  factorization of a matrix  $A+E$ , where  $E$  is a non-negative diagonal matrix. In this section, we briefly review their method. To be consistent with the remainder of the paper, we restate their algorithm in terms of the

Cholesky ( $LL^T$ ) decomposition. This does not change any of the important properties of the algorithm that we discuss.

At each iteration, the algorithm of Gill, Murray, and Wright first selects the maximum (in absolute value) diagonal element in the remaining principal submatrix  $A_j$ , and pivots it to the top left position by interchanging its row and column with the pivot ( $j^{th}$ ) row and column, respectively. Then, if  $A_j$  is now the pivoted principal submatrix, with

$$A_j = \begin{bmatrix} \alpha_j & a_j^T \\ a_j & A_j \end{bmatrix}, \quad (4.1)$$

where  $\alpha_j$  is the diagonal element in the pivot column and  $a_j$  is the remainder of the pivot column, the elements of the next principal submatrix  $A_{j+1}$  are computed by

$$A_{j+1} = \hat{A}_j - \frac{a_j a_j^T}{\alpha_j + \delta_j}. \quad (4.2)$$

The value of  $\delta_j$  at each iteration is chosen to be the smallest non-negative number such that

$$\frac{0 \leq \|a_j\|_\infty}{\alpha_j + \delta_j} \leq \beta,$$

where  $\beta > 0$  is an a priori bound selected to minimize a worst case bound on  $\|E\|_\infty$ . If  $\alpha_j < 0$  and this value of  $\delta$  is less than  $-2\alpha_j$ , then  $\delta_j = -2\alpha_j$  instead.

What remains to be described is the choice of  $\beta$ . Let  $\xi$  = the maximum magnitude of the off-diagonal elements of the original matrix  $A$ , and  $\gamma$  = the maximum magnitude of the diagonal elements of  $A$ . Gill and Murray [1974] produce an error bound on  $\|E\|_\infty$  as a function of  $\beta$  for their algorithm, and show that it is minimized when  $\beta^2 = \xi / \sqrt{n^2 - 1}$ . For that choice of  $\beta$ ,

$$\|E\|_\infty \leq 2(\sqrt{n^2 - 1} + (n-1))\xi + 2\gamma, \quad (4.3)$$

or roughly

$$\|E\|_\infty \leq 4n\xi + 2\gamma \quad (4.4)$$

for moderate to large  $n$ . However this choice of  $\beta$  may cause positive definite matrices  $A$  to be

perturbed, so the selection of  $\beta$  is adjusted in order to avoid this. Gill and Murray [1974] also show that the choice  $\beta \geq \sqrt{\gamma}$  guarantees that  $E = 0$  for positive definite  $A$ . Thus their algorithm assigns  $\beta^2$  to be the maximum value of  $\gamma$ ,  $\xi / \sqrt{n^2-1}$ , or machine epsilon. If  $\gamma > \xi / \sqrt{n^2-1}$ , the usual case, then the error bound for this adjusted  $\beta$  becomes

$$\|E\|_{\infty} \leq (n^2+1)\gamma + 2(n-1)\xi + \xi^2/\gamma, \quad (4.5)$$

which is larger than (4.3).

The modified Cholesky factorization algorithm of Gill, Murray, and Wright [1981] has proven to be an effective factorization in the context of optimization algorithms, and as will be seen in Section 6, does quite a good job of fulfilling the four goals stated at the beginning of Section 3 (The cost of the algorithm is approximately  $n^2$  comparisons, and  $O(n)$  arithmetic operations, more than the standard Cholesky factorization.) It should be noted that while the diagonal pivoting employed by the algorithm of Gill, Murray, and Wright does not affect the analysis described above, it is very important to its good practical performance.

There appear to us to be two important ways in which the algorithm of Gill, Murray, and Wright [1981] might still be improved. First, the bounds (4.3) and particularly (4.5), which are attained by the algorithm for particular matrices  $A$ , are far from optimal, as will be discussed in Section 5. Secondly, the results of Section 6 show that in practice, the value of  $\|E\|_{\infty}$  produced by the algorithm is sometimes many times too large. The new method described in Section 5 primarily attempts to improve upon the algorithm of Gill, Murray, and Wright in these two regards.

## 5. The New Modified Cholesky Factorization

Our new modified Cholesky factorization algorithm incorporates two main new techniques. The first involves using Gershgorin Circle Theorem bounds to determine the elements in the non-negative diagonal matrix  $E$  that is added to an indefinite matrix  $A$  in order to make it positive definite. The second is a new technique for assuring that one does not perturb an already positive definite matrix, i.e. that  $E=0$  if  $A$  is positive definite. In Section 5.1 we describe the new technique that uses Gershgorin bounds to decide how much to add to the diagonal, and show that it leads to an improved upper bound on  $\|E\|_\infty$ . In Section 5.2 we describe the new technique for assuring that a positive definite matrix is not perturbed, and show that unlike the strategy of Gill, Murray, and Wright [1981], it can be incorporated into a modified Cholesky decomposition algorithm without causing the bound on  $\|E\|_\infty$  to grow significantly. In Section 5.3 we describe our full new algorithm, which integrates these two techniques, discuss its theoretical properties, and give a simple example comparing it to the method of Gill, Murray, and Wright [1981].

### 5.1 Using Gershgorin Circle Theorem bounds to determine the amounts to add to the diagonal

In this section, we introduce our basic strategy for choosing a non-negative diagonal matrix  $E$  such that  $A+E$  is positive semi-definite. (The exposition and theory are cleaner if we allow the possibility that  $A+E$  is positive *semi*-definite; the changes to assure that it is strictly positive definite are small in practice and theory, and are described in Section 5.3.) The strategy described in this section may result in  $E$  having some positive elements even if  $A$  is positive definite; the modifications we make to avoid this are described in Section 5.2.

The Gershgorin Circle Theorem states that if  $A \in \mathbb{R}^{n \times n}$  is a symmetric matrix with eigenvalues  $\lambda_1 \leq \dots \leq \lambda_n$ , then each  $\lambda_i \in \{G_1 \cup G_2 \cup \dots \cup G_n\}$ , where

$$G_i = [A_{ii} - \sum_{j=1}^n |A_{ij}|, A_{ii} + \sum_{j=1}^n |A_{ij}|] \triangleq [\text{Glow}_i, \text{Gup}_i], i = 1, \dots, n. \quad (5.1.1)$$

Thus, since  $A - \lambda_1 I$  is positive semi-definite, an upper bound on the amount that must be added to the diagonal of  $A$  to make  $A+E$  positive semi-definite is

$$\text{Maxadd-GCT} \triangleq \max \{0, -\text{Glow}_i\}. \quad (5.1.2)$$

An objective of the new modified Cholesky factorization is to find  $E$  for which  $A+E$  is positive semi-definite and for which we can guarantee

$$\|E\|_\infty \leq \text{Maxadd-GCT}, \quad (5.1.3)$$

at least in the case when we are not concerned about perturbing a positive definite matrix. This bound is easily achieved as indicated by the following lemma and theorem. Note that since, using the notation of Section 4,

$$\text{Maxadd-GCT} \leq \gamma + (n-1)\xi, \quad (5.1.4)$$

(5.1.3) is guaranteed to be stronger than (4.3).

**Lemma 5.1.1.** Let  $A \in R^{n \times n}$  have the Gershgorin Circle Theorem bounds  $G_i, i=1, \dots, n$  given in (5.1.1). Denote  $A = \begin{bmatrix} \alpha & a^T \\ a & \hat{A} \end{bmatrix}$ , where  $\alpha \in R$ ,  $a \in R^{n-1}$ ,  $\hat{A} \in R^{(n-1) \times (n-1)}$ . Let  $\bar{A} = \hat{A} - \frac{aa^T}{\alpha+\delta}$  have Gershgorin Circle Theorem bounds  $\bar{G}_i, i=2, \dots, n$ , where

$$\bar{G}_i = [\bar{A}_{ii} - \sum_{j=1}^{n-1} |\bar{A}_{ij}|, \bar{A}_{ii} + \sum_{j=1}^{n-1} |\bar{A}_{ij}|] \triangleq [\bar{\text{Glow}}_i, \bar{\text{Gup}}_i], i = 2, \dots, n.$$

Then if

$$\delta \geq \max \{0, \|a\| - \alpha\}, \quad (5.1.5)$$

$$\bar{G}_i \subseteq G_i, i=2, \dots, n.$$

**Proof.** Note that (5.1.5) guarantees  $\alpha+\delta \geq 0$ , with equality possible only if  $a=0$ . If  $a=0$ , we may

assume that we set  $\bar{A} = A$  so that the lemma is trivially true. For the remainder of the proof, we assume  $\alpha + \delta > 0$ .

Let us again use the convention that the subscripts of the vector  $a$  are  $i = 2$  through  $n$ , so that  $a_i = A_{i1}$ ,  $i=2, \dots, n$ . Then we have

$$\text{row } i \text{ of } \bar{A} = \text{row } i \text{ of } A - \frac{a^T a_i}{\alpha + \delta}, \quad i = 2, \dots, n.$$

Thus

$$\left| \sum_{j=2}^n |\bar{A}_{ij}| - \sum_{j=2}^n |A_{ij}| \right| \leq \frac{(\|a\|_1 - |a_i|) |a_i|}{\alpha + \delta}. \quad (5.1.6)$$

Also,

$$\bar{A}_{ii} - A_{ii} = \frac{|a_i|^2}{\alpha + \delta}. \quad (5.1.7)$$

Combining (5.1.6) and (5.1.7), recalling that the term  $A_{i1} = a_i$  is present in  $G_i$  but not in  $\bar{G}_i$ , and using  $\delta \geq \|a\|_1 - \alpha = -Glow_1$ , we get

$$\begin{aligned} \bar{Glow}_i - Glow_i &\geq |a_i| - \frac{\|a\|_1 |a_i|}{\alpha + \delta} \\ &= \frac{|a_i|}{\alpha + \delta} (\delta + (\alpha - \|a\|_1)) = \frac{|a_i|}{\alpha + \delta} (\delta + Glow_1) \geq 0, \end{aligned}$$

$i = 2, \dots, n$ . Similar calculations show that

$$\bar{Gup}_i - Gup_i \leq -\frac{|a_i|}{\alpha + \delta} (\delta + Glow_1 + 2|a_i|) \leq 0.$$

Thus  $\bar{G}_i \subseteq G_i$ .  $\square$

Lemma (5.1.1) shows that the choice (5.1.5) causes the Gershgorin intervals to contract. Thus it is almost immediate that if we make this choice with equality at each iteration of the modified Cholesky factorization, we will satisfy (5.1.3).

**Theorem 5.1.2.** Let  $A \in R^{n \times n}$  have the Gershgorin Circle Theorem bounds (5.1.1), and let  $Maxadd-GCT$  be defined by (5.1.2). Suppose that at each iteration of the modified Cholesky factorization, the remaining principal submatrix  $A_j \in R^{(n+1-j) \times (n+1-j)}$  is given by (4.1), ( $A_1 = A$ ),

$$\delta_j = \max\{0, \|a_j\|_1 - \alpha_j\} , \quad (5.1.8)$$

and  $A_{j+1} \in R^{(n-j) \times (n-j)}$  is calculated by (4.2). Let  $E = \text{diag}\{\delta_1, \dots, \delta_n\}$ . Then  $A+E$  is positive semi-definite and (5.1.3) is true. Furthermore, if any diagonal pivoting strategy is used at each iteration (i.e. rows and columns  $i$  and  $j$  are swapped for some  $i > j$ ), (5.1.3) remains true.

**Proof.** The proof is almost immediate from Lemma 5.1.1. Let  $(G^j)_i$ ,  $i=j, \dots, n$  denote the Gershgorin interval obtained from row  $i$  of  $A_j$ , and let  $(G^j low)_i$  denote the lower bound of  $(G^j)_i$ . From Lemma 5.1.1, the choice (5.1.8) assures that

$$(G^{j+1} low)_i \subseteq (G^j low)_i , \quad 1 \leq j \leq i \leq n . \quad (5.1.9)$$

From (5.1.8), (5.1.9), and (5.1.2),

$$\delta_j \leq -(G^j low)_j \leq -Glow_j \leq Maxadd-GCT .$$

This completes the proof of the first part of the theorem. Since diagonal pivoting of a symmetric matrix only permutes its Gershgorin intervals but does not alter them, and since Lemma 5.1.1 and the above part of this proof make no use of the ordering of the Gershgorin intervals, the theorem is unaffected by any diagonal pivoting strategy.  $\square$

Our algorithm makes one further modification to the strategy (5.1.8) for selecting  $\delta_j$ . It is that we require the amount that is added to the diagonal at iteration  $j$  to be at least as great as the greatest amount that has been added to the diagonal at any previous iteration. That is,

$$\delta_j = \max\{0, \|a_j\|_1 - \alpha_j, \delta_{j-1}\} . \quad (5.1.10)$$

It is straightforward that Theorem 5.1.2 remains true with (5.1.10) in place of (5.1.8), because by induction this choice still satisfies (5.1.3), and trivially it still satisfies (5.1.5).

The rationale for this modification is as follows. At any iteration, suppose  $\delta_j$  given by (5.1.10) is larger than that given by (5.1.8) i.e.  $\max\{0, ||a_j||_1 - \alpha_j\} < \delta_{j-1}$ . Then the new choice (5.1.10) doesn't change the value of  $||E||_\infty$  at this point in the algorithm, because  $\delta_j = \delta_{j-1}$ . It may cause subsequent values of  $\delta_i$  to be smaller, however, because it results in a larger  $\alpha_j + \delta_j$  and hence a smaller multiple of  $a_j a_j^T$  is subtracted from  $A_j$ , which means that  $A_{j+1}$  has larger or identical eigenvalues than it would have using (5.1.8). This reasoning does not imply that the final value of  $||E||_\infty$  will be smaller using (5.1.10) than using (5.1.8), but it makes this seem likely, and in practice the modification appears to be helpful in some cases and virtually never harmful.

The total additional work required by the modifications to the Cholesky factorization described so far in this section is approximately  $n^2/2$  additions, for the computation of  $||a_j||_1$  at each iteration. In comparison, the additional work for the algorithm of Gill, Murray, and Wright [1981] is approximately  $n^2/2$  comparisons, because it computes  $||a_j||_\infty$ .

Finally, as noted in Section 4, it is important in practice to use a diagonal pivoting strategy, even though it does not affect the theoretical results given above. We could simply pivot based on the maximum diagonal element, as is done by Gill, Murray, and Wright [1981]. However, recall that the amount we add to the diagonal at iteration  $j$  will be at least the negative of the lower Gershgorin bound of the pivot row for that iteration. This suggests that we instead select as pivot row (and column) the row (and column) for which the lower limit of the Gershgorin interval is largest. If this Gershgorin bound is positive, then we will not increase  $||E||_\infty$  at this iteration, and the Gershgorin intervals will contract.

This pivoting strategy assumes that the Gershgorin bounds for each remaining row are available at each iteration. This would require a total of approximately  $n^3/2$  additional additions, which is too high. An alternative is to pivot based on the estimates of the Gershgorin bounds that result from the proof of Lemma 5.1.1. If we let  $(g^j)_i$  denote the estimate of the lower bound

of the Gershgorin interval of row  $i$  of  $A_j$ , then from the proof of Lemma (5.1.1),

$$(g^{j+1})_i = (g^j)_i + |(a_j)_i| \left[ 1 - \frac{\|a_j\|_1}{\alpha_j + \delta_j} \right], \quad i=j+1, \dots, n.$$

For the entire algorithm, this requires approximately  $n^2/2$  each additional multiplications and additions. To begin this process, the Gershgorin bounds of the original matrix  $A$  must be calculated, which costs an additional  $n^2$  additions. Thus the total costs of the modifications to the Cholesky factorization discussed in the section are  $2n^2$  additions and  $n^2/2$  multiplications.

We should mention that the strategy for preserving positive definiteness that we discuss in Section 5.2 will often cause the additional costs given in this section to be reduced considerably.

## 5.2 The Strategy for Not Perturbing Positive Definite Matrices

In this section we introduce our strategy for assuring that our modified Cholesky decomposition does not perturb an already positive definite matrix, while still guaranteeing that if the matrix is not positive definite, then the amount that is added to the diagonal is not too large. The strategy is quite simple. We divide our decomposition algorithm into two phases. In the first phase, we apply the standard Cholesky decomposition (the version described in Section 3 where we make a rank-one modification to the remaining submatrix at each iteration) for  $k \geq 0$  iterations, stopping at the first occasion that the next,  $k+1^{\text{st}}$  iteration would cause any diagonal element in the next remaining submatrix  $A_{k+2}$  to become non-positive. At this point we know that the current submatrix  $A_{k+1}$ , as well as the original matrix  $A$ , is not positive definite. We then switch to the second phase, where we apply the modified Cholesky decomposition algorithm described in Section 5.1 for the remaining  $n-k$  iterations of the decomposition.

If the original matrix  $A$  is numerically positive definite, then this strategy results in the normal Cholesky decomposition being performed throughout. If  $A$  is not positive definite, then this strategy results in the normal Cholesky decomposition being performed for  $k \in [0, n-2]$

iterations,

followed by the application of the modified Cholesky decomposition to  $A_{k+1}$ , which results in the Cholesky decomposition of  $A_{k+1} + \hat{E}$  for some non-negative diagonal matrix  $\hat{E}$ . The overall result is the Cholesky decomposition of  $A + E$ , where  $E$  is  $\hat{E}$  augmented with zeroes in the first  $k$  diagonal positions (modulo pivoting).

The crucial question is "how large is  $\|\hat{E}\|_\infty$ , and hence  $\|E\|_\infty$ ?". Section 5.1 gives a bound for  $\|\hat{E}\|_\infty$  that depends on the sizes of the elements of  $A_{k+1}$ . In Theorem 5.2.1, we show that our two-phase strategy assures that no element in  $A_{k+1}$  has grown by more than the value of the largest diagonal element in  $A$ . This in turn means that our decomposition still achieves a good bound on  $\|E\|_\infty$  in terms of the original matrix  $A$ .

**Theorem 5.2.1.** Let  $A \in R^{n \times n}$ , and let  $\gamma = \max\{\|A_{ii}\|, 1 \leq i \leq n\}$ ,  $\xi = \max\{\|A_{ij}\|, 1 \leq i < j \leq n\}$ . Suppose we perform the standard Cholesky decomposition as described in Section 3 for  $k \geq 1$  iterations, yielding the remaining principal submatrix  $A_{k+1} \in R^{(n-k) \times (n-k)}$  (whose elements are denoted  $(A_{k+1})_{ij}$ ,  $k+1 \leq i, j \leq n$ ), and let  $\hat{\gamma} = \max\{\|(A_{k+1})_{ii}\|, k+1 \leq i \leq n\}$  and  $\hat{\xi} = \max\{\|(A_{k+1})_{ij}\|, k+1 \leq i < j \leq n\}$ . Then if  $(A_{k+1})_{ii} \geq 0, k+1 \leq i \leq n$ , then  $\hat{\gamma} \leq \gamma$  and  $\hat{\xi} \leq \xi + \gamma$ .

**Proof:** Let  $A = \begin{bmatrix} B & C^T \\ C & F \end{bmatrix}$ , where  $B \in R^{k \times k}$ ,  $C \in R^{(n-k) \times k}$ ,  $F \in R^{(n-k) \times (n-k)}$ . After  $k$  iterations of the Cholesky factorization, the first  $k$  columns of the Cholesky factor  $L$  have been determined;

denote them by  $\begin{bmatrix} \bar{L} \\ M \end{bmatrix}$  where  $\bar{L} \in R^{k \times k}$  is triangular and  $M \in R^{(n-k) \times k}$ . Then

$$B = \bar{L} \bar{L}^T, C = M \bar{L}^T, \text{ and } F = M M^T + A_{k+1}. \quad (5.2.1)$$

From (5.2.1),  $F_{ii} = \|M_{row i}\|_2^2 + (A_{k+1})_{ii}$ ,  $k+1 \leq i \leq n$ , so that from  $F_{ii} \leq \gamma$  and  $(A_{k+1})_{ii} \geq 0$ ,

$$\|M_{row i}\|_2^2 \leq \gamma. \quad (5.2.2)$$

Thus for any off-diagonal element of  $A_{k+1}$ , (5.2.1), (5.2.2) and the definition of  $\xi$  imply

$$(A_{k+1})_{ij} \leq F_{ij} - (M_{row i})(M_{row j})^T \leq \xi + \gamma. \quad (5.2.3)$$

which shows  $\hat{\xi} < \xi + \gamma$ . Also for all the diagonal elements of  $A_{k+1}$ ,  $(A_{k+1})_{ii} \geq 0$ , (5.2.1) and the definition of  $\gamma$  imply

$$0 \leq (A_{k+1})_{ii} \leq F_{ii} \leq \gamma. \quad (5.2.4)$$

which shows  $\hat{\gamma} < \gamma$  and completes the proof.  $\square$

We note that the result of Theorem 5.2.1 is independent of the diagonal pivoting strategy that is used. We also note, however, that the technique of proof of Theorem 5.2.1 actually shows that the largest off-diagonal element in  $A_{k+1}$  is at most equal to the largest off-diagonal in  $F$  plus the largest diagonal in  $F$ , where  $F$ , as defined in the proof of Theorem 5.2.1, is the diagonal submatrix of  $A$  that corresponds to  $A_{k+1}$ . Thus a pivoting strategy that uses the larger diagonal elements as pivots in the first phase will limit the growth in the off-diagonal of  $A_{j+1}$  even more than is indicated by Theorem 5.2.1. Our phase one algorithm pivots the largest remaining diagonal element to the top, and thus is likely to have this effect of further limiting element growth.

The possibility of incorporating this two-phase strategy into the method of Gill, Murray, and Wright [1981] is discussed in the next section.

### 5.3 The Complete New Algorithm

We have now presented all the main parts of our new modified Cholesky decomposition algorithm. An outline of the complete algorithm is given in Algorithm 5.3.1, and a fully detailed description is given in Appendix I. To summarize, the first phase of the algorithm applies the standard Cholesky decomposition, using a diagonal pivoting strategy that pivots the largest remaining diagonal element to the top left. This phase ends when the next iteration of the standard Cholesky decomposition would cause any diagonal element in the remaining submatrix to become non-positive. In the second phase, the modified Cholesky decomposition described in Section 5.1 is applied to the remaining submatrix. This phase determines what to add to the diagonal at each iteration from the lower Gershgorin bound of the pivot row, and pivots based upon estimates of these lower Gershgorin bounds.

Three additional, relatively minor features have been incorporated into Algorithm 5.3.1 to guard against the resultant  $A+E$  being singular or very ill-conditioned. First, the switch to phase two is made when any diagonal element of the remaining submatrix would become less than  $\tau\gamma$ , rather than less than zero as is discussed in Section 5.2. Here  $\gamma$  is again the maximum diagonal of  $A$ , and  $\tau$  is a small constant (we choose  $\tau = (\text{macheps})^{1/3}$ ). This means we may perturb a positive definite matrix if its condition number is greater than  $1/\tau$ . Second, in phase two, to assure that  $A+E$  is positive definite rather than positive semi-definite, we set (using the notation of Section 5.1) each

$$\delta_j = \max \{0, -\alpha_j + \max \{ \|a_j\|_1, \tau\gamma\}, \delta_{j-1}\}$$

where the  $\tau\gamma$  term is new. This causes the bound (5.1.3) on  $\|E\|_\infty$  to increase a tiny bit, to

$$\|E\|_\infty \leq \text{Maxadd-GCT} + \tau\gamma . \quad (5.3.1)$$

but in conjunction with the preceding change, allows us to bound the condition number of  $A+E$ . Finally, at the final iteration of phase two, when only a  $2\times 2$  submatrix  $A_{n-1}$  remains, we use a different strategy : we calculate the eigenvalues  $\lambda_{lo}$  and  $\lambda_{hi}$  of  $A_{n-1}$ , and if  $\lambda_{lo} < \tau \max \{\lambda_{hi}, \gamma\}$ , we

### Algorithm 5.3.1 -- Modified Cholesky Decomposition

Given  $A \in R^{n \times n}$  symmetric and  $\tau$  (e.g.  $\tau = (\text{macheps})^{1/3}$ ),  
find factorization  $L L^T$  of  $A + E$ ,  $E \geq 0$

$$\gamma := \max_{1 \leq i \leq n} |A_{ii}| ; j := 1$$

(\* Phase One, A potentially positive definite \*)

While  $j \leq n$  do

Pivot on maximum diagonal of remaining submatrix

$$\text{If } \min_{j+1 \leq i \leq n} \left\{ A_{ii} - \frac{A_{ij}^2}{A_{jj}} \right\} < \tau \gamma$$

then go to Phase Two

else perform  $j^{\text{th}}$  iteration of standard Cholesky factorization and increment  $j$

(\* Phase Two, A not positive definite \*)

$k := j - 1$  (\*  $k$  = number of iterations performed in Phase One \*)

Calculate lower Gerschgorin bounds of  $A_{k+1}$

For  $j := k+1$  to  $n-2$  do

Pivot on maximum lower Gerschgorin bound estimate

Calculate  $E_{jj}$  and add to  $A_{jj}$

$$(* E_{jj} = \min \{ 0, -A_{jj} + \max \{ \sum_{i=j+1}^n |A_{ij}|, \tau \gamma \}, E_{j-1,j-1} \} *)$$

update Gerschgorin bound estimates

perform  $j^{\text{th}}$  iteration of factorization

complete factorization of final  $2 \times 2$  submatrix using its eigenvalues

choose  $\delta_{n-1}$  so that the  $l_2$  condition number of  $A_{n-1} + \delta I = 1/\tau$  (we also require  $\lambda_{lo} + \delta_{n-1} \geq \tau \gamma$ ).

This generally gives a smaller value of  $\delta_{n-1}$  than the Gerschgorin circle theorem based strategy would, and in theory it is straightforward to show that

$$|\delta_{n-1}| \leq -\lambda_{lo} + \frac{\tau}{1-\tau} (\lambda_{hi} - \lambda_{lo}) \leq \text{Maxadd-GCT} + \frac{2\tau}{1-\tau} \gamma \quad (5.3.2)$$

since  $-\lambda_{lo} \leq \text{Maxadd-GCT}$  and  $\lambda_{hi} - \lambda_{lo} \leq 2(\text{Maxadd-GCT} + \gamma)$ .

The theoretical properties of our full algorithm are summarized in Theorem 5.3.2.

**Theorem 5.3.2.** Let  $A$ ,  $\gamma$ , and  $\xi$  be defined as in Theorem 5.2.1, suppose we apply the modified

Cholesky factorization algorithm in Appendix I to  $A$ , resulting in the factorization  $LL^T$  of  $A+E$ .

If  $A$  is positive definite and at each iteration,  $|L_{jj}|^2 \geq \tau\gamma$ , then  $E=0$ . Otherwise,  $E$  is a non-negative diagonal matrix, with

$$\|E\|_\infty \leq Gersch + \frac{2\tau}{1-\tau} (Gersch + \gamma) \quad (5.3.3)$$

where  $Gersch$  is the maximum of the negative of the lower Gershgorin bounds of  $A_{k+1}$  that are calculated at the start of Phase Two. If  $k=0$  then

$$Gersch = Maxadd - GCT \leq \gamma + (n-1)\xi \quad (5.3.4)$$

where  $Maxadd - GCT$  is given by (5.1.1-2), otherwise

$$Gersch \leq [n - (k+1)](\gamma + \xi). \quad (5.3.5)$$

**Proof:** Immediate from Theorem 5.1.2, Theorem 5.2.1, and equations (5.3.1-2).  $\square$

It is also possible to produce an upper bound on the condition number of  $A+E$ , of the same sort that is provable for the Gill, Murray, and Wright [1981] algorithm. The key properties needed for this are that  $\|E\|$ , and hence  $\max\{|L_{ii}|\}$ , is bounded above, that  $\min\{|L_{ii}|\}$  is bounded below (by  $\sqrt{\tau\gamma}$ ), and that  $|L_{ij}| < L_{ii}$  for all  $1 \leq j < i \leq n$ . (The final property comes from diagonal pivoting and the look-ahead property in phase one, and from the Gershgorin bound strategy for choosing  $\delta_i$  in phase two.) The bound on the condition number that one can obtain is of mainly theoretical interest, since it is exponential in  $n$ ; the computational results of Section 6 show that the condition number of  $A$  is bounded above by about  $1/\tau$  in practice.

We note that our two phase strategy could also be incorporated into the method of Gill, Murray, and Wright [1981], and that this would result in a significant improvement in their upper bound on  $\|E\|_\infty$ . This could be done by using the same two phase structure, and replacing our phase two by their modified Cholesky decomposition. If this were done, their algorithm could simply choose  $\beta^2 = \xi \sqrt{(n-k)^2 - 1}$  in phase two, rather than the maximum of this quantity and  $\hat{\gamma}$  (where  $\xi$  and  $\hat{\gamma}$  are defined as in Theorem 5.2.2) because it would know that it is dealing with a

non-positive definite matrix. Hence the resultant method would achieve the bounds (4.3-4) if it switched to phase two immediately, and

$$\|E\|_{\infty} \leq 4(n-k)\xi + 2\hat{\gamma} \leq 4(n-k)(\xi + \gamma) + 2\gamma$$

otherwise. This would be a significant improvement over the current bound (4.5), although it is still inferior to (5.3.3-5).

Our new algorithm meets our goal of not significantly increasing the cost of the standard Cholesky decomposition, which is about  $n^3/6$  each additions and multiplications. The additional costs of the modified factorization are  $(n-k)^2$  additions to calculate the Gershgorin bounds of  $A_{k+1}$  at the start of phase two, (where  $k$  is the number of iterations performed in phase one),  $(n-k)^2/2$  additions to calculate the  $L_1$  norms of the pivot rows during phase two, and at most  $(n-k)^2/2$  each multiplications and additions to update the Gershgorin bounds during phase two. In addition there is a small multiple of  $n-k$  additional work. (The strategy for precalculating the new diagonal during phase one, in order when to determine when to switch to phase two, only costs a small multiple of  $n$  operations as long as the precalculated values are stored and used when phase one is continued.) Thus the total additional cost of the modified Cholesky decomposition at most  $2n^2$  additions and  $n^2/2$  multiplications, in the case when phase two is started immediately ( $k=0$ ). In many cases in our experience,  $k$  is close to  $n$  so the additional costs are very small.

Finally, we include a small example to demonstrate the performance of the new modified Cholesky algorithm. Consider the matrix used by Gill, Murray, and Wright [1981] to illustrate their modified Cholesky factorization,

$$A = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 1 & 3 \\ 2 & 3 & 1 \end{bmatrix}.$$

Our new algorithm will proceed as follows. At the first iteration, no pivoting is performed in phase one, and then the algorithm immediately switches to phase 2 because  $A_{33} - \frac{A_{31}^2}{A_{11}} < 0$ . The

Gerschgorin intervals of  $A$  are

$$[-2, 4], [-3, 5] \text{ and } [-4, 6].$$

The row with the maximum lower Gerschgorin bound is also row 1, so no pivoting is required in this iteration for phase 2 either. The modified Cholesky algorithm then chooses  $\delta = 2 = -($ Gerschgorin lower bound of row 1), and after the elimination step,

$$A_2 = \begin{bmatrix} 2/3 & 7/3 \\ 7/3 & -1/3 \end{bmatrix},$$

and the estimated Gerschgorin bounds are unchanged. The algorithm now enters the final,  $2 \times 2$  submatrix stage. The eigenvalues of  $A_2$  are (-2.2196, 2.5538), so that  $\delta_2 = 2.2196$  and  $\delta_{total} = 2.2196$ . Thus for the new algorithm,

$$E = \begin{bmatrix} 2 & & \\ & 2.22 & \\ & & 2.22 \end{bmatrix}$$

and  $\|E\|_\infty = 2.22$ . This is 1% greater than the most negative eigenvalue of  $A$  which is -2.2109. (If we had continued the Gerschgorin strategy for  $A_2$  rather than use the eigenvalue strategy,  $\delta_2$  would be 2.67.)

Using the same matrix  $A$ , the Gill, Murray and Wright algorithm computes

$$E = \begin{bmatrix} 2.77 & & \\ & 5.01 & \\ & & 2.24 \end{bmatrix},$$

with  $\|E\|_\infty = 5.01$ .

## 6. Computational Results

We have compared the performance of our new modified Cholesky factorization (Algorithm 5.3.1 and Appendix I) to the algorithm of Gill, Murray and Wright [1981] on a number of indefinite test matrices. The measures we used to assess the performance of the algorithms are the ratios  $\|E\|_\infty / |\lambda_1(A)|$ , termed *relative maxadd*, which reflect how well the algorithm has satisfied the goal of adding as little as possible to the diagonal of  $A$ , and the condition numbers of  $A+E$ . We already know that the other two goals stated at the beginning of Section 3, low cost and not disturbing safely positive definite matrices, are satisfied by both algorithms.

We tested both algorithms on matrices of dimension 25, 50 and 75, with eigenvalue ranges of  $[-1, 10000]$ ,  $[-1, 1]$ , and  $[-10000, -1]$ . For each combination of dimension and eigenvalue range, 10 matrices were created. Thus (the same) 90 test problems that were used to test each algorithm. Each test matrix was created by forming the product  $Q_1 Q_2 Q_3 D (Q_1 Q_2 Q_3)^T$ , where each  $Q_i$  is a Householder matrix of the form

$$Q_i = I - \left[ \frac{2}{\|w\|_2^2} w w^T \right],$$

and each component of each  $w$  is randomly generated from a uniform distribution in the range  $[-1, 1]$ . Each  $D$  is a diagonal matrix whose elements were randomly generated from a uniform distribution in the desired eigenvalue range, with the exception that for the set of test matrices with eigenvalue range  $[-1, 10000]$ , one element of  $D$  was generated from the range  $[-1, 0]$ , thus guaranteeing at least one negative eigenvalue in the test matrices of that range.

The *relative maxadds* for the 90 tests of each algorithm are shown in Figures 1A,C,E, 2A,C,E and 3A,C,E in Appendix II. In summary, the *relative maxadds* for the new algorithm were always small, and sometimes considerably superior to those for the Gill, Murray, and Wright algorithm, although this algorithm's performance was also good in most cases. The

*relative maxadds* for the new algorithm ranged from 1.06 to 2.5, and was below 1.71 for all but 5 of the 90 cases. The *relative maxadds* for the Gill, Murray, and Wright algorithm ranged from 1.6 to 77.8, distributed as follows among the various groups of test matrices. For the matrices with eigenvalues in the  $[-1, 10000]$  range, the *relative maxadds* ranged from 2.1 to 5.6. In the  $[-1, 1]$  eigenvalue range, the *relative maxadds* were in the range 4.9 to 77.8, and in the final  $[-10000, -1]$  eigenvalue range the *relative maxadds* ranged from 1.6 to 5.1. Comparing on a problem by problem basis, the new algorithm performed from 3.5 to 60.9 times better than the Gill, Murray and Wright method in terms of the *relative maxadd* for the problems with the  $[-1, 1]$  eigenvalue range, and from 1.3 to 4.2 times better for the remaining test cases.

Figures 4A-4I show the *relative maxadds* for the new algorithm only, to illustrate more clearly the how close  $\|E\|_\infty$  is to  $\lambda_1(A)$  for this method. Also included in Figures 4A-4I are the results for a version of the new algorithm that differs only in that bases its pivots at each iteration of phase two upon the actual Gershgorin bounds rather than their estimates. The additional cost of calculating these bounds is about  $(n-k)^3/3$ , or at most  $n^3/3$ , additional additions. The results in Figure 4 show that pivoting on the exact Gershgorin bounds leads to some improvement in the size of *relative maxadd*, but we do not consider the improvements sufficient to warrant the extra cost in general.

The condition numbers of  $A+E$  for the two methods are given in Figures 1B,D,F, 2B,D,F and 3B,D,F in Appendix II. Basically, both methods produced acceptably conditioned matrices in all cases. The conditions numbers for the matrices produced by the new method varied from  $10^1$  to  $10^6$ , whereas the condition numbers for the Gill, Murray and Wright method varied from  $10^1$  to  $10^8$ . The condition numbers for the new method are sometimes directly related to the final step of the algorithm, which, if it increases  $\|E\|_\infty$ , does so by the amount necessary to make the final  $2 \times 2$  submatrix positive definite with condition number  $\tau$ . In our test cases, the tolerance  $\tau$  was  $(macheps)^{1/3}$ , or roughly  $10^{-5.2}$  on the Sun 3/75 used for these tests. This accounts for the condition numbers of almost  $10^6$  in all the cases where the final step increased  $\|E\|_\infty$ .

Decreasing this tolerance generally was found to decrease the condition number, usually without appreciably increasing  $\|E\|_\infty$ .

Interestingly, in the cases where the new algorithm produced the most significant improvements in *relative maxadds*, the test problems with the  $[-1, 1]$  eigenvalue range, it also produced much better conditioned matrices than the Gill, Murray, and Wright algorithm. For this test set, the ratios of the Gill, Murray and Wright condition numbers to the condition numbers of the new algorithm were between  $10^2$  and  $10^4$  for  $n = 25$ , between  $10^4$  and  $10^5$  for  $n = 50$ , and between  $10^5$  and  $10^7$  for  $n = 75$ . For the other two eigenvalue ranges, the ratios of the condition numbers produced by the two algorithms all varied by at most 2 orders of magnitude, with the condition numbers for the new algorithm consistently higher for the test problems in the  $[-1, 10000]$  eigenvalue range, and the Gill, Murray and Wright condition numbers usually higher for the test problems in the  $[-10000, -1]$  range.

Finally, Figures 5A,B in Appendix II contain the test results for a different set of matrices of dimension  $n = 25$  with eigenvalue range  $[-1, 10000]$ . The difference between these test matrices and the ones used in figures 1A,B is that these matrices were created to have at least 3 negative eigenvalues, whereas the original test problems in the  $[-1, 10000]$  range were created with at least 1 negative eigenvalue. What is interesting about the results of this new test set is that on one particular matrix out of the 10, the new algorithm performs significantly *worse* than the Gill, Murray and Wright algorithm. (This phenomenon did *not* occur with the test sets of size 50 or 75 in this range with 3 negative eigenvalues, so we have not included this data). The poor behavior occurred when the algorithm was at the  $(n-4)^{th}$  iteration, so we created a  $4 \times 4$  matrix with similar characteristics that illustrates the problem even more markedly

The matrix

$$A = \begin{bmatrix} 1890.3 & -1705.6 & -315.8 & 3000.3 \\ -1705.6 & 1538.3 & 284.9 & -2706.6 \\ -315.8 & 284.9 & 52.5 & -501.2 \\ 3000.3 & -2706.6 & -501.2 & 4760.8 \end{bmatrix}$$

has eigenvalues -0.378, -0.343, -0.248, and 8242.869. The first few steps performed by the new algorithm are as follows:

1. Interchange row and column 4 with row and column 1, because  $A_{4,4}$  is the maximum diagonal element.
2. Switch to phase 2 because  $A_{3,3} - \frac{A_{3,1}}{A_{1,1}} < 0$ .
3. Calculate the lower Gerschgorin bounds {-1447.3, -3158.8, -1049.4, -3131.4}, and since  $-Glow_3$  is the maximum value, interchange row and column 3 with row and column 1.
4. Add  $(-Glow_{pivotrow}) = 1049.4$  to  $A_{1,1}$ .

At this point in the computation, the new algorithm has already added much more to the diagonal than is necessary to make  $A$  positive definite. From this point on it doesn't increase  $\|E\|_\infty$ , so that the final value of  $\|E\|_\infty$  is 1049.4. On the other hand, the Gill, Murray, and Wright algorithm produces  $\|E\|_\infty = 1.01$ . This behavior occurs because, at the first iteration, the Gill, Murray, and Wright algorithm pivots on the maximum diagonal element and then adds nothing to the diagonal, which after elimination results in a  $3 \times 3$  submatrix all of whose entries have absolute value less than 0.52. This is guaranteed to then lead to a small  $\|E\|_\infty$ . (Indeed, if our algorithm performed the same first step as the Gill, Murray, Wright algorithm and then proceeded as usual, it would produce  $\|E\|_\infty = 0.665$ .)

The essential characteristic of this example is that  $A$  is equal to a large symmetric rank one matrix plus a small indefinite matrix. Thus, if nothing is added to  $A_{1,1}$  at the first iteration, the remaining submatrix after the elimination has very small elements, and  $\|E\|$  is small. The Gill, Murray, and Wright algorithm will usually outperform ours on matrices of this type. We

have experimented with modifications to our algorithm that perform well for this case, but all of them resulted in degradation of our algorithm's performance in other cases. Since the case only occurred once in the 120 test cases discussed in this section, we would hope that it is not common in practice.

## 7. Conclusions

We have presented a new modified Cholesky factorization algorithm that does a good job of meeting the objectives outlined at the start of Section 3. It is based upon two new techniques, the use of Gershgorin circle theorem bounds to decide how much to add to the diagonal, and the use of a two phase structure to differentiate between positive definite and non-positive definite matrices. Given a symmetric matrix  $A$ , the factorization produces a Cholesky factorization of a positive definite matrix  $A+E$ . Its cost is at most  $2n^2$  additions and  $n^2/2$  multiplications more than the standard Cholesky factorization, and it does not perturb safely positive definite matrices  $A$ . Its theoretical bound on  $\|E\|$  is a factor of  $n$  lower than for the Gill, Murray, and Wright [1981] method. In computational tests on non-positive definite matrices, it virtually always produces a smaller  $\|E\|$  than the method of Gill, Murray, and Wright, and the conditioning of  $A+E$  is always quite acceptable. On the class of test problems where the Gill, Murray, and Wright algorithm had the most difficulty, those with eigenvalue range [-1,1], the decreases in  $\|E\|$  and in the condition number of  $A+E$  are both substantial.

We have not tested the effect of substituting our new modified Cholesky factorization for that of Gill, Murray, and Wright in optimization algorithms. The most common optimization test problems have small  $n$  and few if any indefinite iterations, so probably there would be little effect on these. The new algorithm might make a difference on problems where  $n$  is larger and there is some indefiniteness. In our opinion, the biggest advantage of the new method for optimization

purposes is its improved theoretical bound on  $\|E\|$  and the corresponding reduction in  $\|E\|$  that has been observed in practice. These properties guard against overflows during the factorization, and against steps  $(A+E)^{-1}\nabla f(x)$  that are far too small.

In addition, the new algorithm leads to an easy implementation of trust region methods for optimization, because  $\|E\|$  is generally within a factor of 1.5 of the smallest eigenvalue  $\lambda_1(A)$  of  $A$ . By first calculating  $E$ , then replacing  $A$  with  $A + (\|E\|)I$  if  $E \neq 0$ , and then using the trust region method for positive definite matrices, one will usually get the solution to the exact, possibly indefinite trust region problem without using any other special provisions for dealing with non-positive definite matrices. We have already used the factorization successfully in this context. If there are other computational algorithms where a crude estimate of the most negative eigenvalue of a matrix is useful, either by itself or as a starting estimate of some iterative procedure, then this factorization may provide a good way to find it.

#### Acknowledgement

We thank Eric Van Vleck for performing the early computational experiments with the new method.

**8. References**

J. E. Dennis Jr. and R. B. Schnabel [1983], *Numerical Methods for Nonlinear Equations and Unconstrained Optimization*, Prentice-Hall, Englewood Cliffs, New Jersey.

P. E. Gill and W. Murray [1974], "Newton-type methods for unconstrained and linearly constrained optimization", *Mathematical Programming* 28, pp. 311-350.

P. E. Gill, W. Murray, and M. H. Wright [1981], *Practical Optimization*, Academic Press, London.

G. A. Shultz, R. B. Schnabel, and R. H. Byrd [1985], "A family of trust region based algorithms for unconstrained minimization with strong global convergence properties", *SIAM Journal on Numerical Analysis*, 22, pp. 47-67.

### Appendix I -- Complete Modified Cholesky Decomposition Algorithm

Given  $A \in R^{n \times n}$  symmetric (stored in lower triangle) and  $\tau$  (e.g.  $\tau = (\text{macheps})^{1/3}$ ),  
find factorization  $L L^T$  of  $A + E$ ,  $E \geq 0$

```

phaseone := true
γ := max |A_ii| for i from 1 to n
j := 1
(* Phase One, A potentially positive definite *)
While j ≤ n and phaseone = true do
    (* Pivot on maximum diagonal of remaining submatrix *)
    i := index of max |A_ii| for j from 1 to n
    if i ≠ j, switch rows and columns i and j of A
    If min_{j+1 ≤ i ≤ n} { |A_{ii} - A_{ij}|^2 / |A_{jj}| } < τγ
        then phaseone := false (* go to Phase Two *)
        else (* perform jth iteration of factorization *)
            L_{jj} = √A_{jj} (* L_{jj} overwrites A_{jj} *)
            For i := j + 1 to n do
                L_{ij} := A_{ij} / L_{jj} (* L_{ij} overwrites A_{ij} *)
                For k := j + 1 to i do
                    A_{ik} := A_{ik} - L_{ij} * L_{kj}
            j := j + 1
    (* end Phase One *)

```

(\* Phase Two, A not positive definite \*)

If phaseone = false then  
 $k := j - 1$  (\*  $k$  = number of iterations performed in Phase One \*)  
(\* Calculate lower Gerschgorin bounds of  $A_{k+1}$  \*)

For i := k + 1 to n do  
 $g_i := \sum_{j=k+1}^{i-1} |A_{ij}| + \sum_{j=i+1}^n |A_{ji}| - A_{ii}$

(\* Modified Cholesky Decomposition \*)

For j := k + 1 to n - 2 do  
 (\* Pivot on maximum lower Gerschgorin bound estimate \*)

$i := \text{index of } \max_{j \leq i \leq n} \{ g_j \}$   
 if  $i \neq j$ , switch rows and columns i and j of A

(\* Calculate  $E_{jj}$  and add to diagonal \*)

$normj := \sum_{i=j+1}^n |A_{ij}|$   
 $\delta (* = E_{jj} *) = \min \{ 0, -A_{jj} + \max \{ normj, \tau \gamma \}, \delta_{\text{prev}} \}$   
 if  $\delta > 0$  then

$A_{jj} := A_{jj} + \delta$   
 $\delta_{\text{prev}} := \delta$  (\*  $\delta_{\text{prev}}$  will contain  $\|E\|_\infty$  \*)

(\* update Gerschgorin bound estimates \*)

If  $A_{jj} \neq normj$  then

$temp := \frac{normj}{A_{jj}} - 1$

for i := j + 1 to n do

$g_i := g_i + |A_{ij}| * temp$

(\* perform jth iteration of factorization \*)

same code as in Phase One

(\* final  $2 \times 2$  submatrix \*)

$$\lambda_{lo}, \lambda_{hi} := \text{eigenvalues of } \begin{bmatrix} A_{n-1,n-1} & A_{n,n-1} \\ A_{n,n-1} & A_{n,n} \end{bmatrix}$$

$$\delta := \max \{ 0, -\lambda_{lo} + \tau * \max \{ \frac{1}{1-\tau} (\lambda_{hi} - \lambda_{lo}), \gamma \}, \delta_{prev} \}$$

if  $\delta > 0$  then

$$A_{n-1,n-1} := A_{n-1,n-1} + \delta$$

$$A_{n,n} := A_{n,n} + \delta$$

$$\delta_{prev} := \delta$$

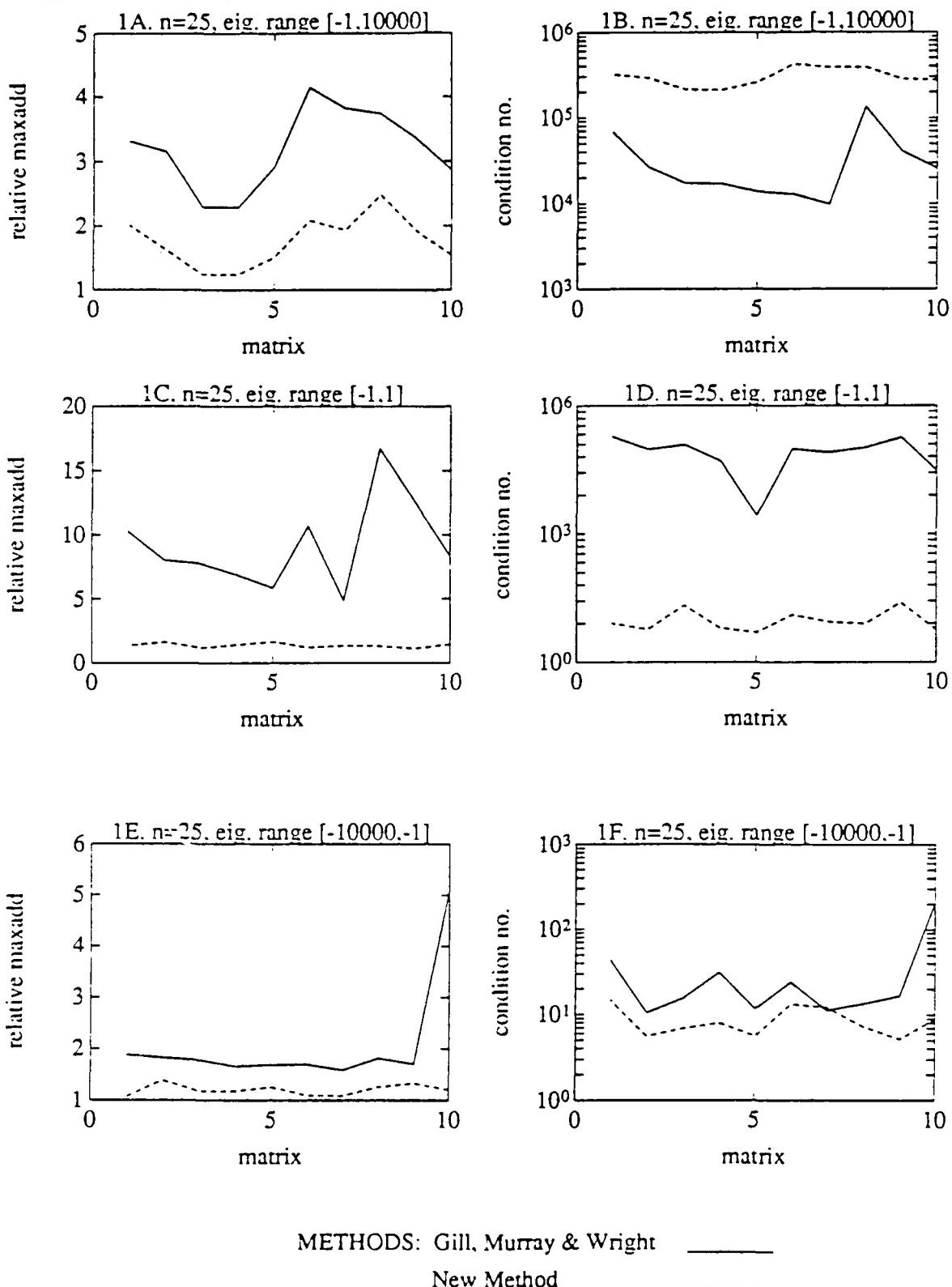
$$L_{n-1,n-1} := \sqrt{A_{n-1,n-1}} \quad (* \text{ overwrites } A_{n-1,n-1} *)$$

$$L_{n,n-1} := A_{n,n-1} / L_{n-1,n-1} \quad (* \text{ overwrites } A_{n,n-1} *)$$

$$L_{n,n} := (A_{n,n} - L_{n,n-1}^2)^{1/2} \quad (* \text{ overwrites } A_{n,n} *)$$

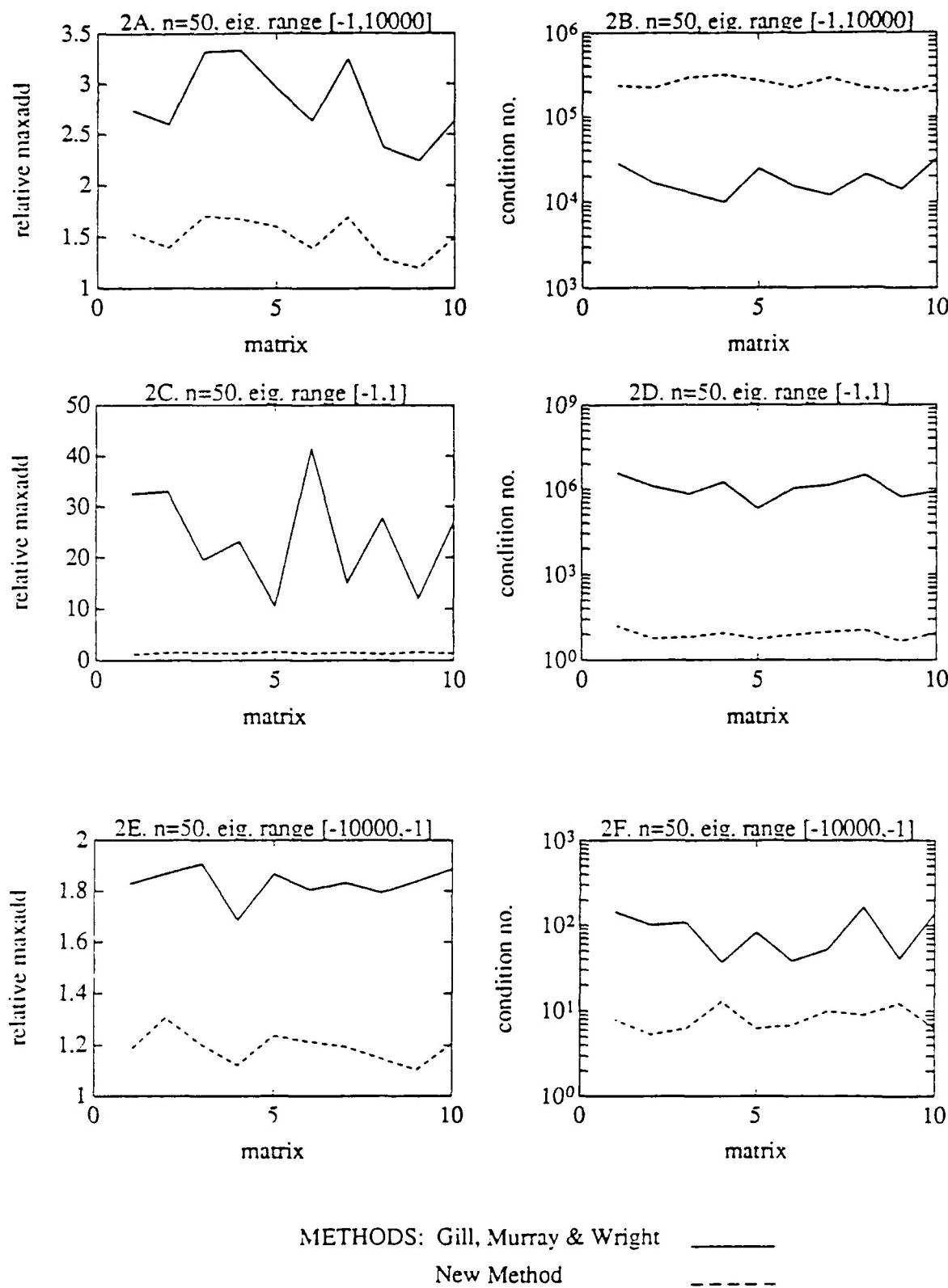
(\* End Phase Two \*)

## Appendix II -- Computational Results



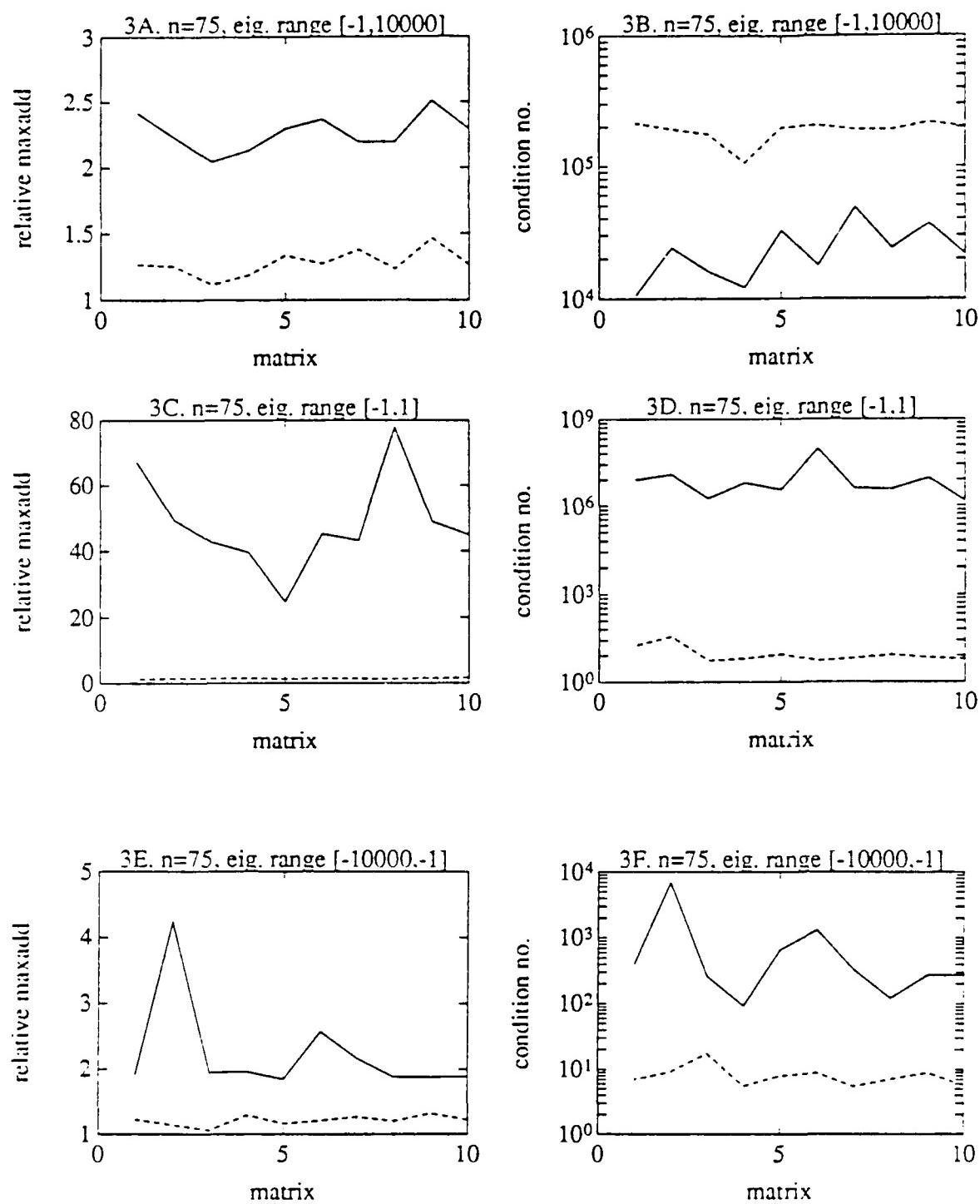
note: relative maxadd = (maximum added to diagonal) / (- smallest eigenvalue)

Figure 1 -- Performance of Existing and New Methods on 10 Indefinite Matrices with  $n=25$



note: relative maxadd = (maximum added to diagonal) / (- smallest eigenvalue)

Figure 2 -- Performance of Existing and New Methods on 10 Indefinite Matrices with  $n=50$

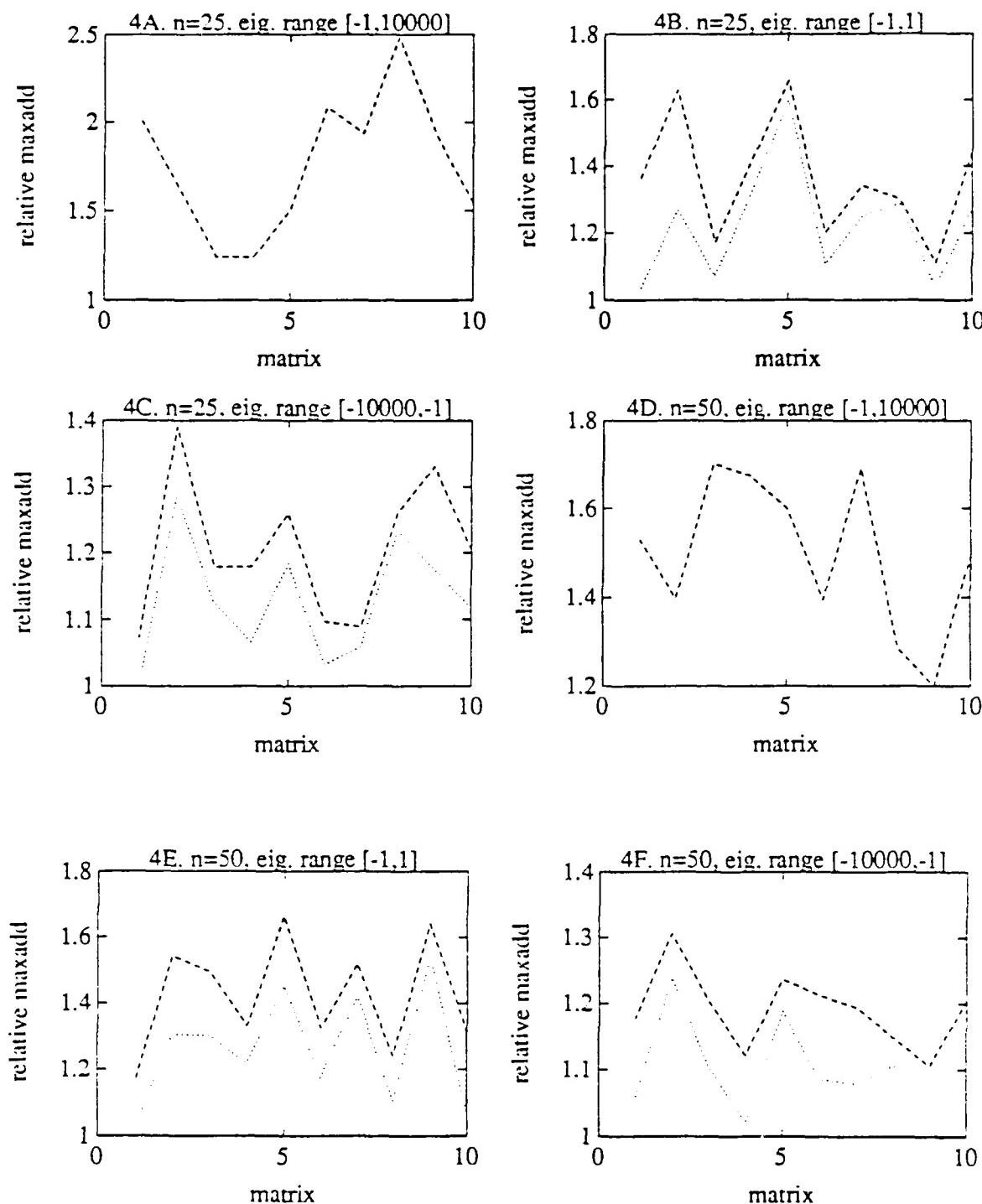


METHODS: Gill, Murray & Wright

New Method

note: relative maxadd = (maximum added to diagonal) / (-smallest eigenvalue)

Figure 3 – Performance of Existing and New Methods on 10 Indefinite Matrices with n=75



METHODS: New Method

New Method with  $O(n^{**}3)$  pivoting .....

(note: two curves are identical in Figures 4 A,D )

Figure 4, Part I – Relative Maxadds for Two Versions of the New Method

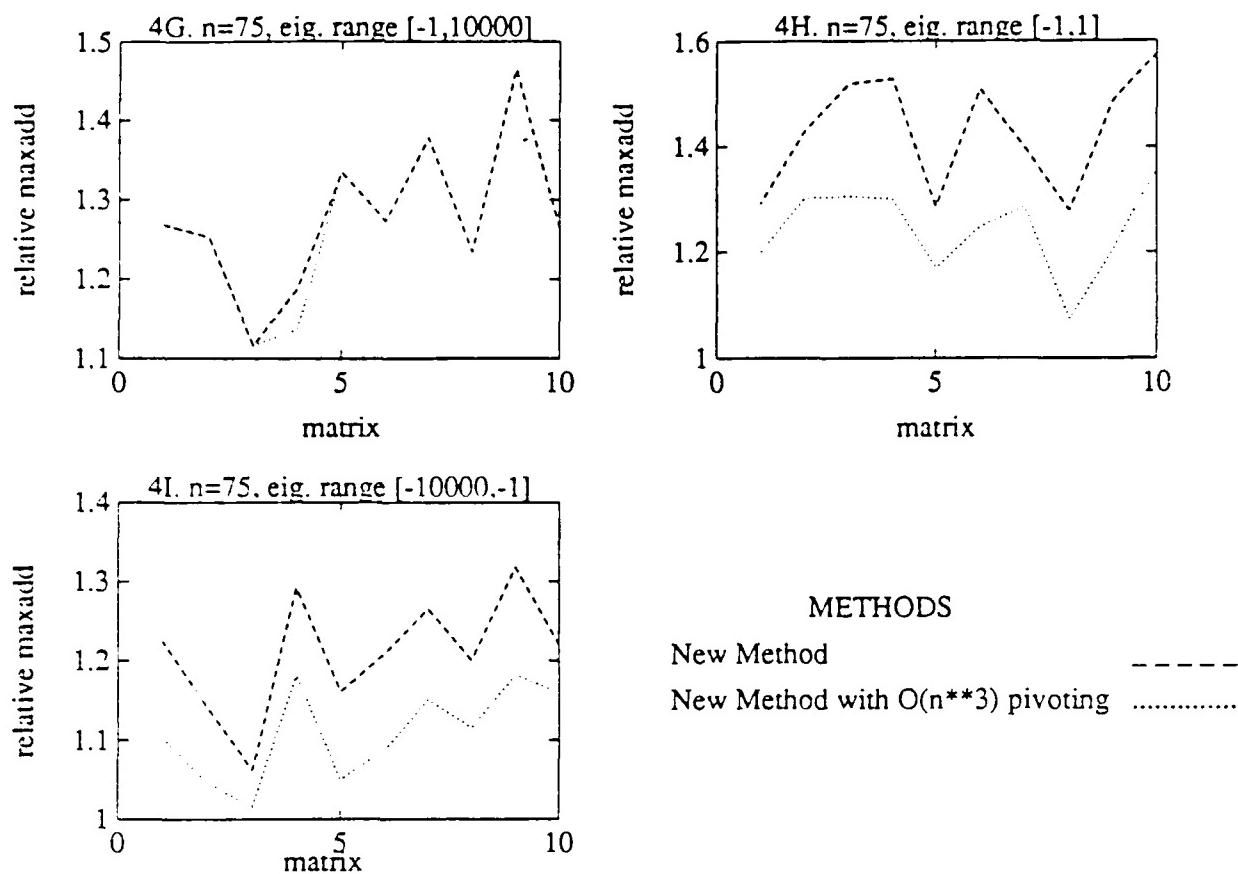
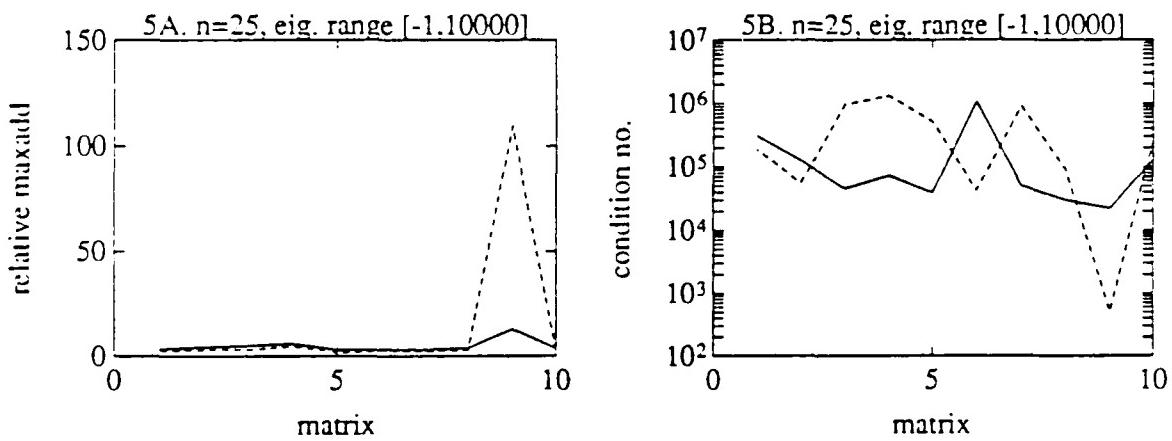


Figure 4, Part II -- Relative Maxadds for Two Versions of the New Method

#### METHODS

New Method

New Method with  $O(n^{**}3)$  pivoting



#### METHODS

Gill, Murray & Wright

New Method

Figure 5 -- Performance of Existing and New Methods on a Test Set with 3 Negative Eigenvalues

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

ADA205174

## REPORT DOCUMENTATION PAGE

1. REPORT SECURITY CLASSIFICATION Unclassified		15. RESTRICTIVE MARKINGS			
2. SECURITY CLASSIFICATION AUTHORITY		16. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited			
20. DECLASSIFICATION/DOWNGRADING SCHEDULE					
4. PERFORMING ORGANIZATION REPORT NUMBER(S) CU-CS-415-38		5. MONITORING ORGANIZATION REPORT NUMBER(S)			
6a. NAME OF PERFORMING ORGANIZATION University of Colorado	6b. OFFICE SYMBOL <i>(If applicable)</i>	7a. NAME OF MONITORING ORGANIZATION U.S. Army Research Office			
6c. ADDRESS - City, State and ZIP Codes Campus Box B-19 Boulder, CO 80309	7b. ADDRESS - City, State and ZIP Codes Post Office Box 12211 Research Triangle Park, NC 27709-2211				
8a. NAME OF FUNDING, SPONSORING ORGANIZATION	8b. OFFICE SYMBOL <i>(If applicable)</i>	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER DAAL03-88-K-0086			
10c. ADDRESS - City, State and ZIP Codes	10. SOURCE OF FUNDING NOS.				
	PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.		
11. TITLE <i>(Include security classification)</i> A New Modified Cholesky Factorization					
12. PERSONAL AUTHORITY Robert B. Schnabel and Elizabeth Eskow					
13a. TYPE OF REPORT Technical	13b. TIME COVERED FROM 88/05/15 TO 89/05/14	14. DATE OF REPORT (Yr., Mo., Day) 88/12/30	15. PAGE COUNT 39		
16. SUPPLEMENTARY NOTATION					
17. COSATI CODES	18. SUBJECT TERMS <i>(Continue on reverse if necessary and identify by block numbers)</i> Cholesky factorization, optimization, non-positive definite, Gerschgorin bounds				
19. ABSTRACT <i>(Continue on reverse if necessary and identify by block numbers)</i>					
The modified Cholesky factorization of Gill and Murray plays an important role in optimization algorithms. Given a symmetric but not necessarily positive definite matrix $A$ , it computes a Cholesky factorization of $A+E$ , where $E=0$ if $A$ is safely positive definite, and $E$ is a diagonal matrix chosen to make $A+E$ positive definite otherwise. The factorization costs only a small multiple of $n^2$ operations more than the standard Cholesky factorization. We present a new algorithm that has these same properties, but for which the theoretical bound on $\ E\ $ is substantially smaller. It is based upon two new techniques, the use of Gerschgorin bounds in selecting the elements of $E$ , and a new way of monitoring positive definiteness. In extensive computational tests on indefinite matrices, the new factorization virtually always produces smaller values of $\ E\ $ than the existing method, without impairing the conditioning of $A+E$ . In some cases the improvements are substantial. The new factorization may prove useful in optimization algorithms.					
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED//UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT <input checked="" type="checkbox"/> OTHER USERS <input type="checkbox"/>		21. ABSTRACT SECURITY CLASSIFICATION Unclassified			
22a. NAME OF RESPONSIBLE INDIVIDUAL Dr. Jagdish Chandra		22b. TELEPHONE NUMBER <i>(Include Area Code)</i> 619/549-0641	22c. OFFICE SYMBOL		